REMARKS

I. Introduction

With the cancellation herein without prejudice of claim 18 and the addition of new claims 19 to 22, claims 7 to 13, 15 to 17, and 19 to 22 are currently pending in the present application. In view of the foregoing amendments and following remarks, it is respectfully submitted that all of the presently pending claims are allowable, and reconsideration is respectfully requested.

II. Rejection of Claims 7 to 10 and 15 to 18 Under 35 U.S.C. § 103(a)

Claims 7 to 10 and 15 to 18 were rejected under 35 U.S.C. § 103(a) as unpatentable over the combination of U.S. Patent No. 6,658,564 ("Smith et al."), U.S. Patent No. 5,941,977 ("Panwar et al."), and U.S. Patent No. 6,374,286 ("Gee et al."). It is respectfully submitted that the combination of Smith et al., Panwar et al., and Gee et al. does not render unpatentable any of claims 7 to 10 and 15 to 18 for at least the following reasons.

As an initial matter, claim 18 has been canceled herein without prejudice, thereby rendering the present rejection with respect to claim 18.

As for the remaining claims, in order for a claim to be rejected for obviousness under 35 U.S.C. § 103(a), the prior art must teach or suggest each element of the claim. *See Northern Telecom, Inc. v. Datapoint Corp.*, 908 F.2d 931, 934 (Fed. Cir. 1990), *cert. denied*, 111 S. Ct. 296 (1990); *In re Bond*, 910 F.2d 831, 834 (Fed. Cir. 1990). To establish a *prima facie* case of obviousness, the Examiner must show, *inter alia*, that there is some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify or combine the references, and that, when so modified or combined, the prior art teaches or suggests all of the claim limitations. M.P.E.P. §2143. In addition, as clearly indicated by the Supreme Court, it is "important to identify a reason that would have prompted a person of ordinary skill in the relevant field to combine the [prior art] elements" in the manner claimed. *See KSR Int'l Co. v. Teleflex, Inc.*, 127 S. Ct. 1727 (2007).

Claim 7, as presented, relates to a method of data processing using a processor including a reconfigurable field of data processing cells and a register, wherein the register has a data stream memory designed as a vector memory, and, as herein amended without prejudice, recites the following:

> *. . . providing a program corresponding to a sequence of compilable high-level language instructions;*

determining, for the reconfigurable field of data processing cells, a set of configurations by execution of which the program is run, wherein each of the configurations is handled as a single instruction;

determining, for each configuration, a respective maximum allowed execution runtime prior to lapse of which the respective configuration is uninterruptible;

executing the configurations; and

during the executing:

storing, in the data stream memory, at least one of the data stream and parts of the data stream; and

for each configuration, monitoring the respective maximum allowed execution runtime in order to interrupt the configuration if the respective maximum allowed execution runtime is exceeded.

With respect to the feature of storing a data stream in a data stream memory during the executing of configurations, the Office Action asserts that Smith et al. at column 4, lines 22 to 33 discloses this feature. Claim 7 has been clarified to indicate that the data stream memory is one designed as a vector memory. Smith et al., on the other hand, merely refer generally to random-access memory devices.

Further, claim 7 provides that there is a respective maximum allowed execution runtime for each configuration. Neither of the cited references, whether considered alone or in combination, disclose or suggest this feature. The Office Action admits that Smith et al. do not disclose this feature and refers instead to Gee et al. as assertedly disclosing this feature. However, as explained in Applicants' Response filed January 24, 2008, Gee et al. refer to operating multiple Java Virtual Machines (JVMs) in separate time slices (partitions) on a single processor, where one master JVM controls the transfers between different JVMs. One JVM of Gee et al. executes a number of applications and threads, and each application or thread will consist of a very large number of instructions. Gee et al. merely indicate that when one JVM (executing a number of applications or threads) exceeds its allotted time, it is terminated. Thus, Gee et al. merely allot fixed periods of time to each JVM that runs plural applications (each consisting of a number of instructions) and threads, but do not determine fixed periods of time for each instruction of the applications. Further Gee et al. are unrelated to configurations.

With respect to application of the allotted times to the configurations of Smith et al., neither of the cited references suggest such an application or indicate how such an application may be implemented. For example, the present application resolves a conflict which arises by it being highly preferable, on the one hand, to have configurations that run as long as possible due to lower configuration overhead, whereas, on the other hand, it is preferred to have a fast interrupt response time. One solution referred to in the present

application is to provide vector memories into which a data stream or parts thereof may be stored during execution, which may allow for fast interrupt response times while it is possible to have configurations that need to run a very long time.

Furthermore, claim 7, as herein amended without prejudice, provides that each of the configurations is handled as a single instruction. The configurations of Smith et al., on the other hand, are classical configurations which are not handled as instructions. For example, a configuration of Smith et al. do not include an execution control so that its execution and control completely depend on an external processor, do not call sub-routines, and are not handled by a operating system.

Moreover, as noted in Applicants' Response of January 24, 2008, a program for a conventional processor is defined by instructions. An instruction for a conventional processor can never exceed a given time. Even if a user experiences the problem of a program that is not responding, the processor itself will nevertheless continue to execute instructions, although this might not lead to a useful or noticeable result of, e.g., a PC being used. Accordingly, there has never been a need to provide a time period for an individual instruction during which it cannot be interrupted but after which it can be interrupted. Indeed, this has never been done. Instead, for a conventional processor, the time needed for the execution of a number of applications as a whole has been monitored, e.g., as in Gee et al.

Even if one skilled in the art would provide for configurations to be handled should correspond to instructions rather than applications, it must still be doubted that the average skilled person would see the need to define a maximum run time for such a configuration. First, this is not necessary for an instruction in a conventional processor. Further, even if the average skilled person would realize that it would be advantageous for a configuration considered in a manner analogous to an instruction to have a restricted time, the average skilled person would still not end up in a more favorable situation with regard to, e.g., debugging, as there would still not be access to something corresponding to state information in a conventional processor. So, there seems to be no use in restricting the run time because there still is no way of debugging the program. Debugging the program becomes possible only if, in addition to the time restriction to the configuration, some state-like information is made available as well. This is done in the present application by storing, in a data stream memory designed as a vector register, at least one of the data stream and parts of the data stream.

Having noted this, it is clear that providing a maximum allowed execution runtime prior to lapse of which a respective configuration, which is handled as a single

instruction, is uninterruptible, as provided for in the context of claim 7, cannot be compared to the termination in Gee et al. of a JVM (executing a number of applications or threads) when it exceeds its allotted time or to the application of the termination of Gee et al. to the configurations of Smith et al.

Panwar et al. do not correct these critical deficiencies of the combination of Smith et al. and Gee et al. For all of the foregoing reasons, the combination of Smith et al., Panwar et al., and Gee et al. does not disclose or suggest all of the features recited in claim 7, as presented, so that the combination of Smith et al., Panwar et al., and Gee et al. does not render unpatentable claim 7.

Claims 8 to 10 and 15 to 17 ultimately depend from claim 7 and are therefore allowable for at least the same reasons set forth above in support of the patentability of claim 7. *In re Fine*, 837 F.2d 1071, 5 U.S.P.Q.2d 1596 (Fed. Cir. 1988) (any dependent claim that depends from a non-obvious independent claim is non-obvious).

Withdrawal of this obviousness rejection is therefore respectfully requested.

III. Rejection of Claim 11 Under 35 U.S.C. § 103(a)

Claim 11 was rejected under 35 U.S.C. § 103(a) as unpatentable over the combination of Smith et al., Panwar et al., Gee et al., and U.S. Patent No. 5,860,119 ("Dockser"). It is respectfully submitted that the combination of Smith et al., Panwar et al., Gee et al., and Dockser does not render unpatentable the present claims for at least the following reasons.

Claim 11 depends from claim 7 and is therefore allowable for at least the same reasons set forth above in support of the patentability of claim 7 since Dockser does not cure the deficiencies noted above with respect to the combination of Smith et al., Panwar et al., and Gee et al.

Withdrawal of this obviousness rejection is therefore respectfully requested.

IV. Rejection of Claims 12 and 13 Under 35 U.S.C. § 103(a)

Claims 12 and 13 were rejected under 35 U.S.C. § 103(a) as unpatentable over the combination of Smith et al., Panwar et al., Gee et al., and U.S. Patent No. 4,041,462 ("Davis et al."). It is respectfully submitted that the combination of Gonion et al., Panwar et al., Gee et al., and Davis et al. does not render unpatentable either of claims 12 and 13 for at least the following reasons.

Claims 12 and 13 ultimately depend from claim 7 and are therefore allowable for at least the same reasons set forth above in support of the patentability of claim 7 since Davis et al. do not cure the deficiencies noted above with respect to the combination of Smith et al., Panwar et al., and Gee et al.

Withdrawal of this obviousness rejection is therefore respectfully requested.

V. New Claims 19 to 22

New claims 19 to 22 have been added herein. It is respectfully submitted that new claims 19 to 22 do not add any new matter and are fully supported by the present application, including the Substitute Specification. Claims 19 to 22 ultimately depend from claim 7 and are therefore allowable for at least the same reasons set forth above in support of the patentability of claim 7.

VI. Conclusion

In light of the foregoing, it is respectfully submitted that all of the presently pending claims are in condition for allowance. Prompt reconsideration and allowance of the present application are therefore earnestly solicited.

Respectfully submitted,

Dated: August 19, 2008

By: /Aaron Grunberger/
Aaron Grunberger
Reg. No. 59,210

KENYON & KENYON LLP
One Broadway
New York, New York 10004
(212) 425-7200

CUSTOMER NO 26646